CECS 551: Advanced Artificial Intelligence (Spring 2018)

Homework #2
Due 2/26/2018

---

1. You need to submit a report in hard-copy before lecture and your code to BeachBoard.
2. Hard-copy is due in class before lecture and electronic copy is due 2:50PM on BeachBoard on the due date.
3. Unlimited number of submissions are allowed on BeachBoard and the latest one will be graded.

---

1. (10 points) [Cross-entropy error measure]

   (a) More generally, if we are learning from ± 1 data to predict a noisy target $P(y|x)$ with candidate hypothesis h, show that the maximum likelihood method reduces to the task of finding h that minimizes

   $$E_{in}(w) = \sum_{n=1}^{N} [\![y_n = +1]\!] ln \frac{1}{h(x_n)} + [\![y_n = -1]\!] ln \frac{1}{1 - h(x_n)} \tag{1}$$

   (b) For the case $h(x) = \theta(w^\top x)$, argue that minimizing the in-sample error in part (a) is equivalent to minimizing the one in $E_{in}(w) = \frac{1}{N} \sum_{n=1}^{N} ln(1 + e^{-y_n w^\top x_n})$.

   For two probability distributions $\{p, 1-p\}$ and $\{q, 1+q\}$ with binary outcomes, the cross-entropy (from information theory) is

   $$p \, log \frac{1}{q} + (1-p) log \frac{1}{1-q} \tag{2}$$

   The in-sample error in part (a) corresponds to a cross-entropy error measure on the data point (x_n, y_n), with $p = [\![y_n = +1]\!]$ and $q = h(x_n)$

2. (20 points) Recall the objective function for linear regression can be expressed as

   $$E(w) = \frac{1}{N} \|Xw - y\|^2,$$

   as in Equation (3.3) of LFD. Minimizing this function with respect to w leads to the optimal w as $(X^T X)^{-1} X^T y$. This solution holds only when $X^T X$ is nonsingular. To overcome this problem, the following objective function is commonly minimized instead:

   $$E_2(w) = \|Xw - y\|^2 + \lambda \|w\|^2,$$

   where $\lambda > 0$ is a user-specified parameter. Please do the following:

   (a) (10 points) Derive the optimal w that minimize $E_2(w)$.

(b) (10 points) Explain how this new objective function can overcome the singularity problem of $X^T X$.

3. (35 points) In logistic regression, the objective function can be written as

$$E(w) = \frac{1}{N} \sum_{n=1}^{N} \ln \left(1 + e^{-y_n w^T x_n}\right).$$

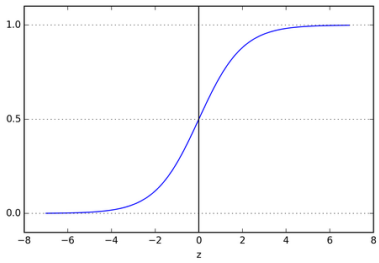
Please

(a) (10 points) Compute the first-order derivative $\nabla E(w)$. You will need to provide the intermediate steps of derivation.

(b) (10 points) Once the optimal w is obtain, it will be used to make predictions as follows:

$$\text{Predicted class of } x = \begin{cases} 1 & \text{if } \theta(w^T x) \geq 0.5 \\ -1 & \text{if } \theta(w^T x) < 0.5 \end{cases}$$

where the function $\theta(z) = \frac{1}{1+e^{-z}}$ looks like



Explain why the decision boundary of logistic regression is still linear, though the linear signal $w^T x$ is passed through a nonlinear function θ to compute the outcome of prediction.

(c) (5 points) Is the decision boundary still linear if the prediction rule is changed to the following? Justify briefly.

$$\text{Predicted class of } x = \begin{cases} 1 & \text{if } \theta(w^T x) \geq 0.9 \\ -1 & \text{if } \theta(w^T x) < 0.9 \end{cases}$$

(d) (10 points) In light of your answers to the above two questions, what is the essential property of logistic regression that results in the linear decision boundary?

4. (35 points) **Logistic Regression for Handwritten Digits Recognition**: Implement logistic regression for classification using gradient descent to find the best separator. The handwritten digits files are in the "data" folder: train.txt and test.txt. The starting code is in the "code" folder. In the data file, each row is a data example. The first entry is the digit label ("1" or "5"), and the next 256 are grayscale values between -1 and 1. The 256 pixels correspond to a 16×16 image. You are expected to implement your solution based on the given codes. The only file you need to modify is the "solution.py" file. You can test your solution by running "main.py" file. Note that code is provided to compute a two-dimensional feature (symmetry and average intensity) from each digit image; that is, each digit image is represented by a two-dimensional vector before being augmented with a "1" to form a three-dimensional vector as discussed in class. These features along with the corresponding labels should serve as inputs to your logistic regresion algorithm.

(a) (15 points) Complete the ***logistic_regression()*** function for classifying digits number "1" and "5".

(b) (5 points) Complete the ***accuracy()*** function for measuring the classification accuracy on your training and test data.

(c) (5 points) Complete the ***thirdorder()*** function to transfer the features into 3rd order polynomial Z-space.

(d) (10 points) Run "main.py" to see the classify results. As your final deliverable to a customer, would you use the linear model with or without the 3rd order polynomial transform? Briefly explain your reasoning.

Deliverable: You should submit (1) a hard-copy report (along with your write-up for other questions) that summarizes your results before the lecture and (2) the "solution.py" file to the BeachBoard.

Note: Please read the "Readme.txt" file carefully before you start this assignment. Please do NOT change anything in the "main.py" and "helper.py" files when you program.